Remote Process Control using a Mobile Phone (2)

part 2*: programming and use

The SMS Chip checks incoming text for a password, switches outputs interactively or at preset times, controls an LCD and sends status information back to any mobile phone.

In the first part of this article we looked at the circuit design and hardware for this project and now we turn our attention to the PCB layout, the software and the text message structure.

A look at the component placement diagram in **Figure 1** shows that you will not need any special soldering skills to complete the circuit board. No wire links are required and all the ICs (except the voltage regulator) are fitted into sockets. All external connections are made via pinheaders or connectors mounted around the edge of the PCB. The LED arrays D1 and D2 can be replaced by groups of eight individual LEDs but if you use high-efficiency types it will be necessary to substitute 1 kΩ or 1.5 kΩ resistor networks for R1 and R2. In all cases it will still be necessary to use the ACT type buffer specified in the parts list for IC3 and IC4.

The voltage regulator IC8 requires a heat sink fitted along the top surface of the PCB. Be sure to fit an insulation sheet between the PCB and the heat-sink to ensure that none of the PCB tracks are short-circuited.

The on/off switch S4 can be simply mounted on the PCB or to a front panel if the unit is mounted in an enclosure.

Three screws with pillars are used to fix the LCD onto the SMS ExBo interface board. The fourth mounting hole does not line up with the hole in the interface board but a screw and pillar should be fitted to the LCD to give it additional support. Leave the bottom end of the pillar free. Non-conducting washers are used under the heads of the

SMS Chip features

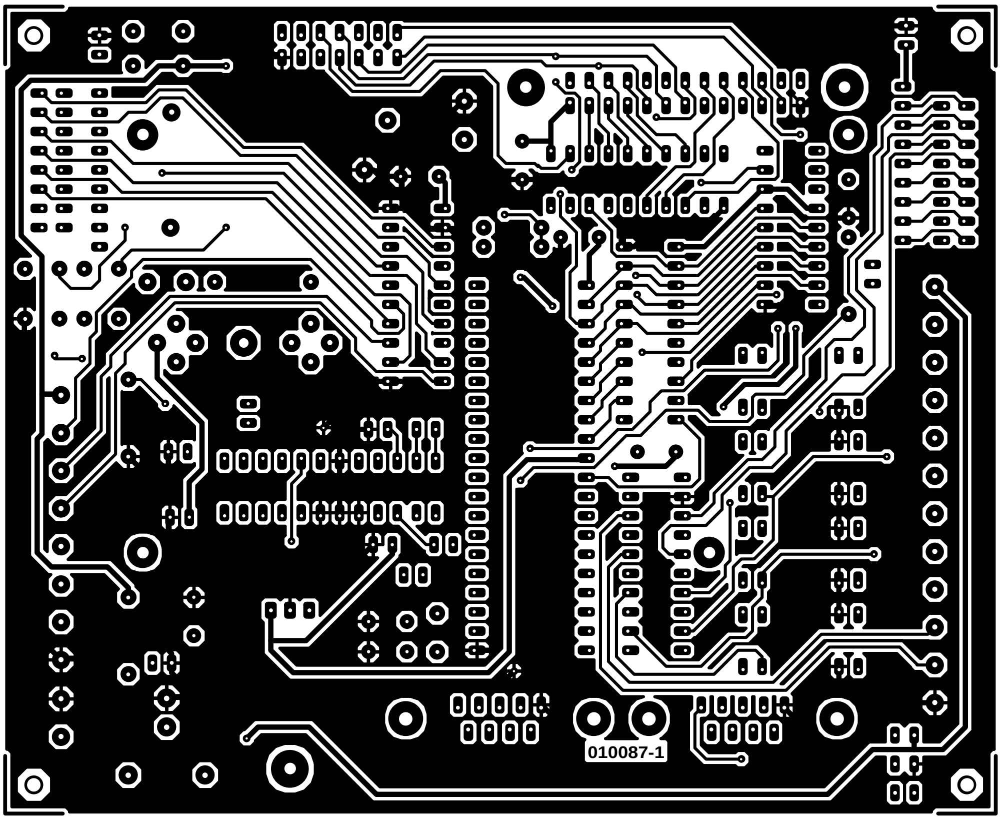
The SMS Chip* is implemented in an AT89LS8252 from Atmel (a programmable 8051 derivative)

- 8 kByte On-Chip Flash-Programmable memory
- 2 kByte On-Chip Flash-Data memory
- 16 digital port pins (Port P1 and Port P2), individually programmable as inputs or outputs.
- Serial Interface 1 for communication with the mobile and loading the configuration data.
- Serial interface 2 UART implemented in software (9600 baud) for communication with an external µC system (PLC, PC)
- Software Real Time Clock (RTC) without battery back-up.
- Optional external hardware Real Time Clock (RTC) with battery back-up (IC6).
- Control for an alphanumeric LC Display to display text messages.
- Connector for external circuit interface (Port P0).
- Package outline DIP-40 or PLCC-44

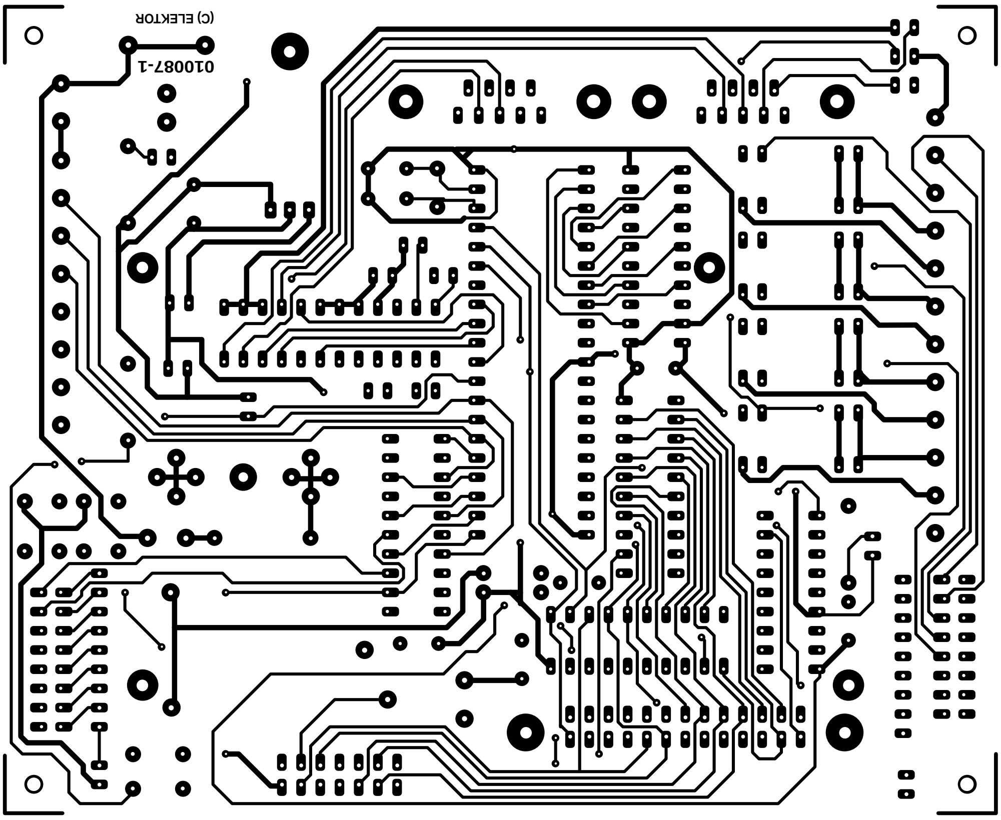
* The SMS Chip is exclusively available from:
J. Engelmann & U. Schrader
Im Schmiedehofe 14
D-31035 Despetal-Barfelde
Germany.
Tel. (+49) 5182 903520
Fax (+49) 5182 903530
Website: www.engelmann-schrader.de
Email: info@engelmann-schrader.de

---

* This instalment could not be included in the February 2002 issue as planned because of late delivery of the German original, and the extra time needed to localize various project related files for use in English speaking countries. Our apologies for any inconvenience caused. *Ed.*



Component side



Solder side

Figure 1a. PCB layout and …

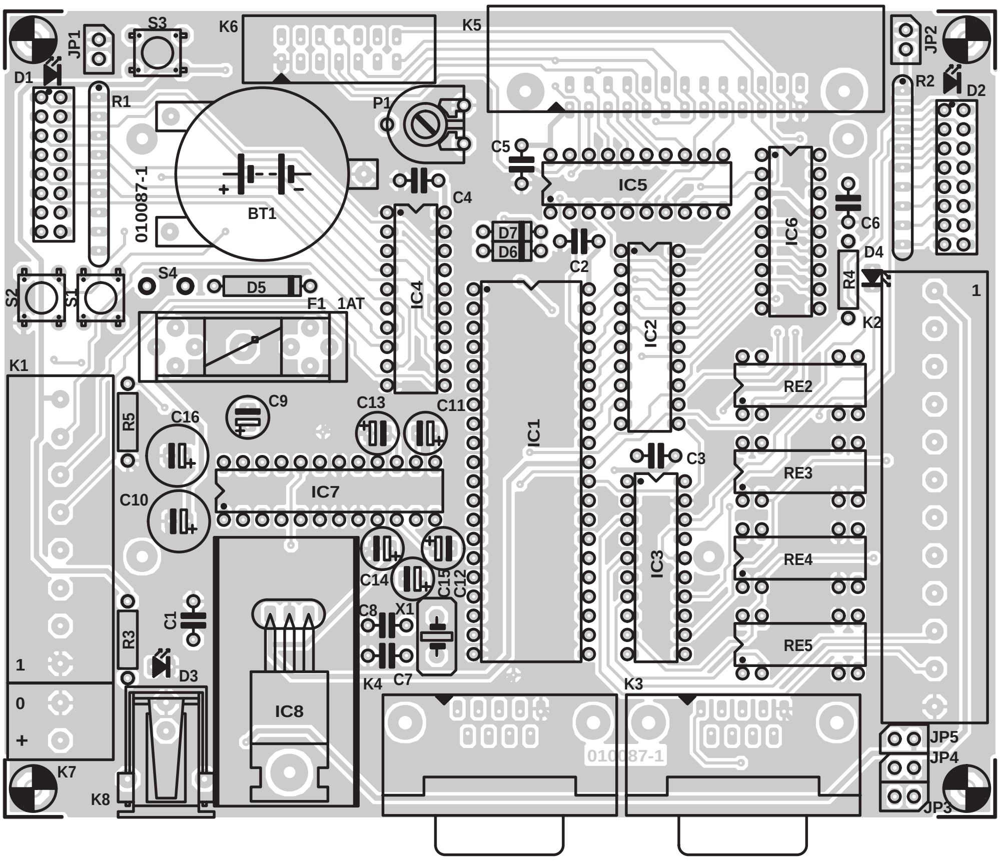


Figure 1b. … component placement of the SMS ExBo.

three screws fixing the pillars to the interface board to prevent possible short-circuits between any of the PCB tracks.

The Data Cable

Connection of the adapter card to a mobile can be accomplished using an off-the-shelf data cable for the Siemens '35 series GSM phone. These cables are usually used to connect the mobile to a PC serial port. Using such a cable has one big disadvantage in our application because it does not allow the mobile to be re-charged from the SMS ExBo. No provision is made in the cable to supply a charging current for the mobile's battery. If you used this cable then it would be necessary to periodically disconnect the mobile from the SMS ExBo and charge up the batteries with the phone charger unit. Using this option it is essential that both jumpers J4 and J5 are removed and jumper J3 fitted (see **Table 2** in the first part of this article). The preferred alternative is to make a slight modification to the data cable wiring. Firstly dismantle the plug housing at the phone end of the data cable (see Figure 2) and solder an additional wire (same length as the data cable) to pin 3. Lead this wire out through the cable entry opening and then re-assemble the connecter. Fit an in-line 1 Ω, 1 W resistor in this wire and connect the free end of the wire to + 5 V on the SMS ExBo (pin 1 of connector K2 would be suitable).

The charging current can be up to 500 mA so it is important not to stint on the mains adapter unit specified in the parts list. Fit jumper J3 but do not fit jumpers J4 and J5, these are only intended for future use with other types of mobiles.

SMS Chip operation

With the PCB finished and the data cable made up we can turn our attention to the SMS chip and how the SMS ExBo is controlled. The SMS Chip must first be initialised before it will recognise any commands sent in the SMS text.

SMS Chip initialisation

The SMS chip needs to be configured with information such as the mobile telephone number, pre-loaded text messages and so on before it can be used. The configuration data can be entered using a standard text editor program. See the accompanying text box giving an example of the configuration data.

The configuration data is stored as ASCII in the file `SMSCHIP.CFG` and the program used to send this data to the SMS ExBo is the DOS program `SMSCONFG.EXE`. To load configuration data it is first necessary to remove jumper J3 (jumpers J4 and J5 should not be fitted) and then fit a standard serial (female to female) 'null modem' cable from COM1 of the PC to K3 of the SMS ExBo. Press the reset button S3 and the 'mobile ready' indicator (D4) labelled 'power' on the circuit diagram will flash to show that the SMS Chip is waiting 10 s for the configuration data to be sent. Two clicks on the `SMSCONFG.EXE` file will automatically open up a DOS window and show transfer of the configuration data.

Some users may need to use the COM2 port, in which case you should call the program as before from the DOS window, but this time use the command

```
SMSCONFG.EXE /2
```

to start the program. Once the transfer has been completed, close the DOS window and press reset button S3. D4 will again blink for 10 s (it has already been configured so just ignore D4) and then the SMS Chip looks for a mobile connected to K3. Disconnect the null-modem cable from the PC to the SMS ExBo and connect the mobile to K3 using the data cable. Jumper J3 can now be fitted.

A short description of the configuration data file together with valid parameters is given in a text box in this article but for a more detailed description refer to the SMS Chip handbook.

The SMS Chip instruction set

After the SMS Chip is configured it is ready to accept commands. Altogether there are twelve commands that the SMS Chip recognises and these can easily be entered using SMS ‘texting’ (see **Table 1**). All of the commands have the same structure: first comes the password then the command and finally the message terminator. Detailed descriptions of all the commands are included in the handbook for the SMS Chip, free to download from the *Elektor Electronics* web site. Any visitor to the web site will also notice an additional program associated with this project written in C51. The SMS-1.c and its associated hex file SMS-1.hex. This program allows anyone who has built the 537 ‘Lite’ computer featured in the January/February 2000 editions of Elektor Electronics to use it as an external computer at the remote site. In this configuration the

COMPONENTS LIST

Resistors:
R1,R2 = 8-way 330Ω SIL array
R3 = 1kΩ5
R4 = 1kΩ8
R5 = 4kΩ7
P1 = 10kΩ preset

Capacitors:
C1, C2, C4,C5,C6 = 100nF (5mm lead pitch)
C3 = not fitted
C7,C8 = 27pF
C9 = 10µF 63V radial
C10 = 100µF 25V radial
C11-C15 = 1µF 16V radial
C16 = 100µF 10V radial

Semiconductors:
D1,D2 = LED, 3mm (8 pcs) or array *
D3, D4 = LED, green, high efficiency
D5 = 1N4002
D6 = BAT48
D7 = 1N4148
IC1 = AT89(L)S8252-24PC in DIP40 case. Programmed, available from Engelmann&Schrader
IC2 = 74HCT573
IC3,IC4 = 74ACT240*
IC5 = GAL16V8, programmed, Publishers’ order code **010087-31**
IC6 = RTC72421
IC7 = MAX207 or ADM207EAN
IC8 = 7805 with ICK35SA heatsink *

Miscellaneous:
BT1 = CR2032 Lithium button cell with PCB mount holder
F1 = fuse, 1A, slow, with PCB mlunt holder
JP1-JP5 = jumper
K1 = 8-way PCB terminal block, lead pitch 5mm
K2 = 12-way PCB terminal block
K3,K4 = 9-way sub-D plug (male), PCB mount, angled pins *
K5 = 26-way boxheader, angled pins
K6 = 14-way boxheader
K7 = 2-way PCB terminal block, lead pitch 5mm
K8 = mains adaptor socket, PCB mount
RE1 = not fitted
RE2-RE5 = 5V PCB mount relay, 1 make or 1 changeover contact, with flyback diode (e.g., Meder 1A72-12D 5V or Siemens V23100-V4305-C010 or Conrad Electronics #504580) *
S1,S2,S3 = pushbutton with 1 make contact
S4 =switch, on/off *
X1 = 11.0592MHz quartz crystal
LC Display = alphanumeric display with HD44780 controller (or compatible), e.g., 4 lines of 20 characters
PCB, Publishers’ order code **010087-1**
Disk, project software, Publishers’ order code **010087-11**
Battery eliminator (mains adaptor) 9V / 1A

* See text

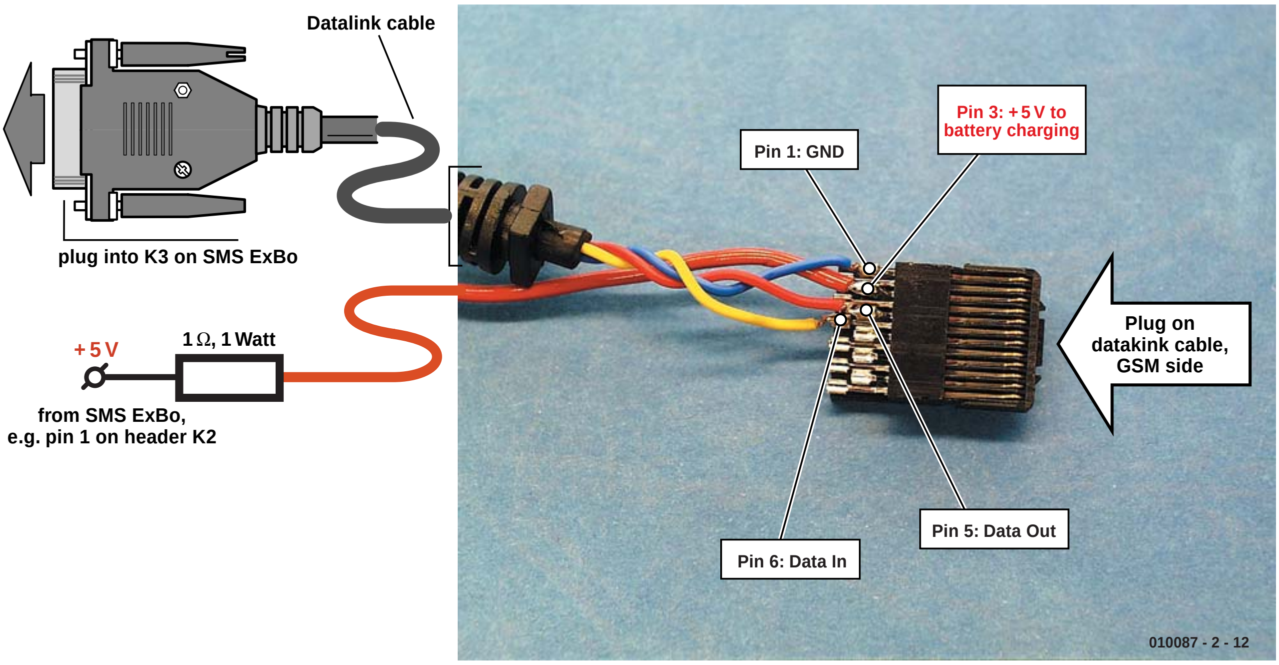


Figure 2. Modification of an off-the-shelf data cable to allow in-circuit charging of the mobiles battery.

Free Software

To download any of these files go to the Free Downloads area of our website at www.elektor-electronics.co.uk

SMSCHIP.CFG	Configuration data
SMSCONFG.EXE	load program for smschip.cfg
SMS-1.c	C control program for the 537 ‘Lite’ computer
SMS-1.hex	Hex file for the above program sms-1.c

537 ‘Lite’ computer will receive SMS text from the SMS ExBo and send its own text message back out through the same path.

Putting it all to use

When a message is received the SMS Chip reads all the SMS text and then automatically erases the message from the mobile’s memory. If a standard text message (without any password or configuration commands) is sent to the mobile, the SMS Chip will try to decode it but find nothing of interest in the message so it will just delete it from the mobiles message store.

An important feature of this system is the response time of messages through the GSM network. The time delay between sending a message from one mobile and it being received on the other mobile can vary from a few seconds up to an hour or so. It is therefore important to note that this system will not be suitable to control or monitor time-critical processes! Suitable applications would be for example controlling heating systems or monitoring cold-storage refrigerators.

During set-up and testing you will need to send many messages to the system just to confirm correct operation of the unit. A good alternative to using a mobile phone to send the messages is to use an Internet connected PC. There are several web sites that allow you to send text messages to mobile phones free of charge (try www.lycos.co.uk). This will save not only the strain on your finances but also help avoid that painful medical condition known as ‘texters finger’. These web sites add an advertising slogan either before or after your message but the SMS chip will simply ignore it. It only decodes text following the ‘message start’ marker (the password) and stops decoding when the ‘message end’ symbol (close bracket) is received. All of the text outside these markers will be ignored. A problem could occur if the advertising slogan has been inserted in the middle of your message but this is very unlikely and I’ve never yet found a site that does this. If this were to occur, the SMS chip would recognise that the message has been corrupted and ignore the message completely. A valid message is necessary before any of the outputs are switched.

If you object to the advertising slogans then it will be necessary to find a site that does not use them. However in the world of the non-existent free lunch (TANSTAFL principle) you will have to pay for sending the text. Other sites on the Internet offer an e-mail to SMS gateway. As the name suggests these sites provide a link between the GSM text network and the Internet so that e-mails can be sent and received as text messages on a GSM phone (www.locust.co.uk or www.airmail.co.uk). Additional information on this option can be found in the handbook.

(010087-II)

Table 1. The SMS Chip command instruction set.

(		
“<password>”		
SET	<pin>	// Set the specified pin to 1
or		
SET	<identifier>	// Set the specified pin to 1
RESET	<pin>	// Reset the specified pin to 0
or		
RESET	<identifier>	// Reset the specified pin to 0
OFF	<pin>	// Set the specified pin to 1
or		
OFF	<identifier>	// Set the specified pin to 1
ON	<pin>	// Reset the specified pin to 0
or		
ON	<identifier>	// Reset the specified pin to 0
PULSE	<pin> <duration>	// Output a pulse on the specified pin
or		
PULSE	<identifier> <duration>	// Output a pulse on the specified pin
OUTPUT	<address> <values>	// 8-Bit output from XDATA (8 Bit Address)
INPUT	<address> <count>	// 8 bit input from XDATA (8 Bit Address)
WRITE	“<text>”	// Output to the 2. serial interface
DISPLAY CLS		// Clear LC Display
or		
DISPLAY SCROLL	“<text>”	// Enter Text in the last display line and scroll
or		
DISPLAY	<column> <line> “<text>”	// Enter text giving line and column information
EVENT	<time> DELETE	// Delete time EVENT
or		
EVENT	<time> <address> <count> SINGLE	// Time EVENT occurs once only
or		
EVENT	<time> <address> <count> EVERYDAY	// Time EVENT occurs daily
TIME	<time>	// Current time
REPORT NO		// No report (Default)
or		
REPORT YES		// report
or		
REPORT ERROR		// Report only when an error occurs
)		

Configuration file example.

```
-CONFIG
Time           :1200
Name           :Enterprise
Password       :Scotty
Masterno       :441794711081
Pin  0 I/O     :I
Pin  0 Name    :Warp1
Pin  0 Telefon :441781234567
Pin  0 Text    :Pressure too high!
Pin  1 I/O     :I
Pin  1 Name    :ALARM
Pin  1 Telefon :4512345876
Pin  1 Text    :Romulan on board!
...
Pin  7 I/O     :I
Pin  7 Name    :Pin7
Pin  7 Telefon :441704711081
Pin  7 Text    :Report pin 7!
...
Pin  8 I/O     :O
Pin  8 Name    :Pin8
Pin  8 Telefon :440000000000
Pin  8 Text    :(Pin8)
...
Pin 15 I/O     :O
Pin 15 Name    :Photon24
Pin 15 Telefon :440000000000
Pin 15 Text    :(Pin15)
Event  1 Time  :1200
Event  1 Typ   :E
Event  1 Start :00
Event  1 Count :05
Event  2 Time  :1205
Event  2 Typ   :E
Event  2 Start :10
Event  2 Count :12
Event  3 Time  :1210
...
Event  8 Time  :0000
Event  8 Typ   :O
Event  8 Start :00
Event  8 Count :0
LCD Lines      :04
LCD Char/Lines :20
LCD Addr 0     :00
LCD Addr 1     :64
LCD Addr 2     :20
LCD Addr 3     :84
```

Data header

-CONFIG
Key word indicating start of the configuration data.

Time
Actual time for the software Real Time Clock (RTC) in the SMS chip. Note: no separating colon is entered between hours and minutes.

Name
This allows an identifying name to be allocated to the SMS chip. It will use this name whenever it sends out SMS text.

Password
Every SMS text message sent to the remote mobile must use the password defined here. Any SMS message received without this password will be ignored by the SMS chip.

Masterno
This is the main telephone number that the SMS chip will send its text messages to.

Digital I/O port pins

Pin x I/O
This defines pin x as an input or output: I ≡ Input, O ≡ Output. When the pin is configured as an input, the SMS chip will automatically send out a message when the input level changes from high to low (falling edge). It will only send another message when the input level returns to a high state
Pin x Name
This allows you to assign a name to a pin so that in the text message you can refer to the pin number or the pin name whichever is more meaningful.

Pin x Telefon
The mobile telephone number entered here will be called and will receive the SMS text when a falling edge is detected on pin x. If no number is entered then the mobile with the master number (Masterno) will be called.

Pin x Text
The message (32 characters max) that will be sent out when the change in pin x is detected.

“Pin x Telefon” and “Pin x Text” are only valid if pin x is defined as an input pin.

Internal and timed events

The SMS chip can send system information automatically at predefined times to the mobile phone number (Masterno). A total of eight events are available (x=1 to 8).

Event x Time
The time that the SMS message will be sent.

Event x Typ
Three different types of event are possible:
O (Off): The corresponding event is deactivated.
S (Single): The event will be activated once only at the time specified.
E (Everyday): The event will be activated daily at the time specified.

Event x Start
The start address (decimal) in external memory of data (XDATA range) that will be included and sent out in the SMS text message.

Event x Count
This defines how much data and the start address of the data to be sent out. In the example listing event 2 will send out 12 data bytes in an SMS message from the external memory starting at address 10 at 12:05 every day.

The use of external memory is not covered in this article but the SMS ExBo system bus is available at connector K5 for the user to attach external memory (e.g. dual port RAM) to store this data.

LCD operation

LCD Lines
The number of lines on the LCD.

LCD Char/Lines
Characters per line

LCD Addr 0 to 3
The address of the first character in each of the display lines. The values are given in the corresponding data sheets.